VERIFICATION OF TRANSLATION

I, Yukari Hirano, registered Patent Attorney, whose address is c/o Okada, Fushimi and Hirano, PC, NE Kudan Bldg. 5F, 2-7, Kudan-minami 3-chome, Chiyoda-ku, Tokyo 102-0074, Japan, hereby certify that I am conversant with the Japanese and English languages, and that to the best of my knowledge and belief the accompanying document is a true English translation of Japanese Patent Application No. 2000-74236 (JP 2000-74236).

Signature _____

Date this 19 day of January, 2005.

[Document Name] Patent Application
[Document Number] H100017001
[Date] March 16, 2002
[To] Commissioner, Japan Patent Office
[International Classification] F02D 45/00
[Inventor]
[Address] c/o K.K. Honda Gijutsu Kenkyusho
4-1, Chuo 1-chome, Wako-shi, Saitama, Japan
[Name] Tetsuya YASHIKI
[Inventor]
[Address] c/o K.K. Honda Gijutsu Kenkyusho
4-1, Chuo 1-chome, Wako-shi, Saitama, Japan
[Name] Masanori MATSUURA
[Inventor]
[Address] c/o K.K. Honda Gijutsu Kenkyusho
4-1, Chuo 1-chome, Wako-shi, Saitama, Japan
[Name] Naohiko MIZUO
[Applicant]
[ID] 000005326
[Name] Honda Giken Kogyo Kabushiki Kaisha
[Attorney]
[ID] 100081721
[Patent Attorney]
[Name] Tsuguo OKADA
[Official Fee]
[Account No.] 034669
[Fee] ¥21,000
[Accompanying Documents]
[Object] Specification 1
[Object] Drawings 1
[Object] Abstract 1
[Proof] Yes

[Document Name] Specification

[Title of the invention]

Memory rewriting system for vehicle controller

[Claims]

[Claim 1] A memory rewriting system for a vehicle controller comprising:

a memory area mounted in the vehicle controller and from and to which data can be deleted and written, the memory area storing first security data used to determine the presence of a permission of rewriting to the memory area;

a rewriting device for transferring second security data from an exterior to said vehicle controller; and

rewriting means mounted in the vehicle controller, for deleting the first security data and writing the second security data transferred from the rewriting device.

[Claim 2] The memory rewriting system for a vehicle controller according to claim 1, wherein the second security data is written using a program; and

wherein the program is stored in a memory area which is mounted in the vehicle controller and from or to which data cannot be deleted or written.

[Claim 3] The memory rewriting system for a vehicle controller according to claim 1 or 2, wherein the second security data is arbitrarily set by the rewriting device.

[Claim 4] The memory rewriting system for a vehicle controller according to any of claims 1 to 3, wherein the permission of rewriting with the first security data is provided if an anti-theft system permits an operation of the vehicle.

[Detailed Description of the Invention]

[0001]

[Technical Field to which the Invention Pertains]

The present invention relates to a memory rewriting system for rewriting a program stored in a memory of a vehicle controller with another program transferred from an external rewriting device.

[0002]

[PRIOR ART]

Vehicles are subjected to various types of control by an electronic control unit (hereafter referred to as "ECU"). Such control includes engine-related control for an air fuel ratio, fuel injection amount, and emission as well as body-related control for a power window, an air bag, and an ABS. The ECU provides various types of control for the vehicle based on current conditions and traveling status of the vehicle sensed by various sensors mounted on the vehicle.

[0003]

On the other hand, the vehicle may include an anti-theft system. In general, the anti-theft system electronically checks if an ignition key used by a driver to start the engine is authentic. If it is determined that the key is authentic, the anti-theft system transfers a signal for permitting vehicle operation to the ECU. Thus, until the permission signal is received, the ECU does not allow the engine to start by, for example, stopping fuel injection. If it is determined that the ignition key is not authentic, the driver is judged to be not an authorized person and cannot operate the vehicle.

[0004]

The ECU comprises a central processing unit (CPU), a ROM (Read Only Memory) that stores programs and data to be executed, a RAM (Random Access Memory) which provides a work area for execution and which stores results of computation, and an I/O interface for receiving signals from various sensors and transmitting control signals to various parts of the engine.

[0005]

The ROM often includes a rewritable memory such as a flash memory, an EEPROM, or an EPROM to allow a program or data therein to be rewritten. Japanese Patent Application Laid-Open No. 63-223901 describes a method for changing a program stored in the EEPROM of the ECU in response to a request from an external device with the ECU being mounted on the vehicle.

[0006]

Such a function of changing a program or data stored in a ROM of the ECU makes it necessary to protect them from access from an external device, thus preventing a user or other third parties from rewriting a program or data stored in the ROM without proper authorization. Japanese Patent Application Laid-Open No. 3-238541 describes a vehicle controller for determining that a program or data in a ROM of the ECU is tampered using a check data mechanism. According to the mechanism, check data based on data stored in the ROM are stored beforehand. After shipment of the vehicle, the ECU creates new check data based on the data stored in the ROM. The ECU then compares the new check data with the previously stored check data,

determines that the data have been tampered if they are unequal and turns on the alarm light.

[0007]

A key for releasing the above-mentioned security feature is known only to a manufacturer of a rewriting device under contract to the automobile manufacturer. Thus, only the rewriting device authorized by the automobile manufacturer can use the "key" and change the data stored in the ROM of the ECU of that automobile.

[0008]

A typical procedure for changing a program in the ROM will be described in brief. The above-mentioned key is typically expressed by a certain function, which is provided both in the rewriting device and in the ECU. The rewriting device is connected to the ECU and then uses its own function (i.e., key) to calculate a function value for an arbitrary numerical value transmitted from the ECU. The rewriting device then transfers the function value to the ECU. At the same time, the ECU uses its own function (i.e., key) to calculate a function value for the same numerical value. The ECU compares the function value received from the rewriting device with the function value determined by itself. If they are equal, the ECU releases the security feature. Thus, the rewriting device is permitted to rewrite data stored in the ROM. If they are unequal, then the rewriting device is judged to be not authentic because the rewriting device and the ECU have different functions (keys). Consequently, the security

feature is not released and the rewriting device cannot rewrite the data stored in the ROM.

[0009]

[Problems to be Solved by the Invention]

The key for releasing the security feature, however, is conventionally stored in a non-rewritable area of the ROM in the ECU, so that it is impossible to use the rewriting device to change the key after the vehicle has been shipped. Thus, if the key is accidentally divulged to a user or another third party who is not authorized, a rewriting device other than the authorized one can rewrite the key in the ROM, thereby breaking the security feature.

[0010]

On the other hand, if the vehicle includes an anti-theft system and if a program used to operate the anti-theft system is rewritten, then the anti-theft system would be invalidated. Accordingly, a system for rewriting a program or data stored in the ROM requires higher security than that for the anti-theft system.

[0011]

The present invention solves these problems. An object of the present invention is to provide a memory rewriting system for a vehicle controller which enables, even after shipment of the vehicle, changing of a key for releasing a security feature that prevents a program or data stored in the ROM of the ECU from being tampered. Even if the key has been divulged to a third party who is not authorized, the manufacturer can

use the rewriting device to change the key, thus enabling the security feature to be easily recovered.

[0012]

Another object of the present invention is to provide a memory rewriting system for a vehicle controller which can operate in cooperation with an anti-theft system.

[0013]

[Means to Solve the Problem]

To solve the above problems, a memory rewriting system for a vehicle controller according to claim 1 comprises a memory area mounted in the vehicle controller and from and to which data can be deleted and written, the memory area storing first security data used to determine the presence of a permission of rewriting to the memory area, a rewriting device for transferring second security data from an exterior to the vehicle controller, and rewriting means mounted in the vehicle controller, for deleting the first security data and writing the second security data transferred from the rewriting device.

[0014]

According to the invention set forth in claim 1, even if the security data for determining the presence of the permission of rewriting to prevent data stored in the memory of the vehicle controller from being rewritten illegally is divulged to a third party, the rewriting device can change the security data, thus preventing illegal rewriting to the memory from spreading.

[0015]

According to the invention set forth in claim 2, in the memory rewriting system for a vehicle controller according to claim 1, the second security data is written using a program, and the program is stored in a memory area which is mounted in the vehicle controller and from or to which data cannot be deleted or written.

[0016]

According to the invention set forth in claim 2, the program for rewriting the security data is stored in the unchangeable memory area and is prevented from being tampered by a third party, thereby allowing the security data to be rewritten safely.

[0017]

According to the invention set forth in claim 3, in the memory rewriting system for a vehicle controller according to claim 1 or 2, wherein the second security data is set arbitrarily by the rewriting device.

[0018]

According to the invention set forth in claim 3, the rewriting device can arbitrarily set a new security data, so that the new security data can be flexibly set without being divulged to any third person.

[0019]

According to the invention set forth in claim 4, in the memory rewriting system for a vehicle controller according to any of claims 1 to 3, wherein the permission of rewriting with the first security data is granted if an anti-theft system permits an operation for the vehicle.

[0020]

According to the invention set forth in claim 4, the memory is rewritten only if the anti-theft system permits an operation for the vehicle, so that rewriting by an illegal driver is avoided, thus preventing information on the anti-theft system from being rewritten.

[0021]

[Mode for Carrying out the Invention]

The present invention for rewriting a security program stored in a non-volatile memory of a vehicle controller will be described referring to attached drawings. The present invention, however, is not limited to the system for rewriting the security program but is applicable to various systems for rewriting data stored in a non-volatile memory.

[0022]

FIG. 1 shows an outline of a memory rewriting system according to one embodiment of the present invention. The memory rewriting system comprises an electronic control unit (ECU) 10 mounted on a vehicle 1 and a rewriting device 11. The rewriting device 11 is authorized by the manufacturer of the vehicle 1. The ECU 10 comprises a rewritable ROM (not shown). As shown in the figure, when the rewriting device 11 is connected to the ECU 10 and some appropriate operation to the rewriting device 11 is performed, a security feature for preventing a program or data stored in the ROM of the ECU 10 from being rewritten without proper authorization is released. Thus, the rewriting device is allowed to rewrite the program or data stored in the ROM.

[0023]

Rewriting is executed via serial communication between the ECU 10 and the rewriting device 11. A user can send data for rewriting to the ECU 10 by operating buttons on the rewriting device 11 and/or interacting with a display screen provided on the rewriting device 11. The rewriting device, however, is not limited to the form shown in the figure, but may be of another form having a protocol that enables communication with the ECU 10.

[0024]

FIG. 2 is a functional block diagram showing the entire memory rewriting system according to one embodiment of the present invention. As described above, the memory rewriting system comprises the ECU 10 mounted on the vehicle and the rewriting device 11. The rewriting device 11 is provided outside the ECU 10 and connected thereto via serial communication. Alternatively, parallel communication may be used between the rewriting device 11 and the ECU 10.

[0025]

The ECU 10 comprises a central processing unit 14 (hereafter referred to as a "CPU") including a microcomputer and associated circuit elements, ROMs 16 and 18 which are non-volatile memories and which store programs and data, a RAM 37 (Random Access Memory) which provides a work area for execution and which stores results of computations, and an I/O interface 38 for receiving signals from various sensors 39 and transmitting control signals to various parts of the engine. Signals from various sensors 39 include an engine

rotation speed (Ne), an engine water temperature (Tw), an intake air temperature (Ta), a battery voltage (VB), and an ignition switch (IGSW). Thus, based on a signal input from the I/O interface 38, the CPU 14 invokes a control program and data from the ROMs 16 and 18 to execute computations, and outputs the results to various parts of the vehicle via the I/O interface 38 to control various functions of the vehicle.

[0026]

The ECU 10 also comprises an interface 12. The interface 12 has a protocol for communication with the rewriting device 11 to enable serial communication between the ECU 10 and the rewriting device 11.

[0027]

The rewritable ROM 16 is a memory from and to which stored data can be deleted and new data can be written The rewritable ROM 16 can be, for example, a flash memory or an EEPROM. The non-rewritable ROM 18 can be implemented by specifying a part of the memory area of the rewritable ROM as an unchangeable area, or by using a mask ROM for which data are fixed during manufacturing and from or to which data can subsequently not be deleted or written. Alternatively, the ROM 18 can be implemented with a PROM to which data can be written only once.

[0028]

The ROMs 16 and 18 can be implemented as two memories that are physically separated. Alternatively, the memory area of a single memory may be divided into two areas so that one of the areas is used as a rewritable area, while the other is used as a non-rewritable area. In the latter case, for example,

after a non-rewritable area in which a program or the like is stored has been specified in the EEPROM, a rewritable area is specified with a start address and an end address in the unfilled space of the memory.

[0029]

Now, examples of a form of the ROMs 16 and 18 and CPU will be described with reference to FIG. 3. In this figure, the ROMs 16 and 18 are implemented using a flash memory. FIG. 3(a) shows a form in which the flash memory is provided separately from the CPU. When a rewriting operation mode is entered through communication with the rewriting device, the CPU receives program code from the rewriting device, and invokes a program for rewriting the flash memory with the received program code.

[0030]

On the other hand, FIG. 3(b) shows a form having a built-in flash memory that constitutes one chip in conjunction with the CPU. When the rewriting operation mode is entered in response to a signal from the rewriting device, program codes transferred from the rewriting device is automatically written to the flash memory using a function incorporated in the CPU. The memory rewriting system according to the present invention is applicable to either of the above forms.

[0031]

Referring back to FIG. 2, the rewritable ROM 16 stores a security function f_2. The security function f_2 is an object of rewriting by the rewriting device 11. The security function

f_2 realizes a security feature for preventing the data stored in the ROM 16 from being illegally rewritten.

[0032]

The non-rewritable ROM 18 stores programs for implementing an authentication part 31, a random number generator 33, and a rewriting part 35. The authentication part 31 is responsive to a request for releasing security from the rewriting device 11, and determines whether the rewriting device 11 is authentic using the security function f_2 and a random number R which is generated by the random number generator 33. Using the random number R enables the security feature to be enhanced. If it is determined that the rewriting device is authentic, the authentication part 31 releases the security feature. After that, the rewriting part 35 deletes the security function f_2 and receives a new security function f_3 from the rewriting device 11 to write it into the ROM 16.

[0033]

The rewriting device 11 has a security function f_1 and a new security function f_3. The security function f_1 implements the security feature in cooperation with the security function f_2 stored in the ROM 16 of the ECU 10. If the security function f_2 has not been changed by any third person, the security function f_1 of the rewriting device 11 is the same as the security function f_2 of the ECU 10. In another embodiment, the security functions f_1 and f_2 have a certain relationship.

[0034]

The new security function f_3 is to be stored in the ROM 16 in place of the security function f_2 in order to implement

a new security feature. The new security function f_3 can be created by making certain changes to the current security functions f_1 and f_2. According to one example, the new security function f_3 is a function that has a different expression from the security functions f_1. According to another example, the new security function f_3 is a function that has different constant(s) in the function expression from the security functions f_1. For example, when the functions f_1 and f_2 are $f_1 = f_2 = A \times R + B$ ($A = 10$ and $B = 5$), the new security function f_3 is set such that $f_3 = A + R \times B$ ($A = 10$ and $B = 5$). Alternatively, the values of the constants A and B of the functions f_1 and f_2 may be changed to 5 and 10, respectively.

[0035]

The rewriting device 11 also comprises a security release request part 21, a rewriting request part 23, and a data block assembling part 25, which may be stored in a memory of the rewriting device 11 as programs. The security release request part 21 uses the security function f_1 to request the ECU 10 to release the security feature.

[0036]

The data block assembling part 25 assembles data blocks suitable for serial communication from program code of the security function f_3. The data block assembling part 25 divides the program code of the security function f_3 into a plurality of pieces, each of which having a certain length (for example, 8 bits). An address field is added to each piece of the program code, or each partial program code. The address field includes a leading address of an area in which the partial program code

is to be stored. Thus, when each partial program code is transferred to the ECU, the ECU is informed of a location where the partial program code is to be stored.

[0037]

The rewriting request part 23 serially transfers to the ECU 10 the data blocks representative of the new security function f_3 assembled by the data block assembling part 25 after the security feature has been released.

[0038]

An anti-theft system 81 is connected to the ECU 10 so that the memory rewriting system can exchange information with the anti-theft system 81. The anti-theft system 81 extracts an electronic code from an ignition key inserted into a key cylinder when the engine is to be started and compares the electronic code with a predetermined authorized code to check whether the inserted ignition key is authentic. If it is determined that the ignition key is authentic, the anti-theft system 81 transfers a signal indicative of a permission for engine start to the ECU 10 via an I/O interface 38. In response to receiving this permission signal, the ECU 10 can start an engine.

[0039]

If it is determined that the inserted ignition key is not authentic, the permission signal is not output and the ECU 10 cannot start the engine. In response to the permission signal to the ECU 10, an engine start permission flag which may be stored in the RAM 37 or ROM 16 is set to a value of one. Although the anti-theft system 81 and the ECU 10 are

separately shown in FIG. 2, some of the functions of the anti-theft system 81 may be included in the ECU 10. For example, the authorization of the ignition key may be performed by the ECU 10.

[0040]

The operation of the memory rewriting system shown in FIG. 2 is described with reference to FIGS. 4 and 5. Rewriting operation is initiated, for example, when an operation button of the rewriting device 11 is pressed after the rewriting device 11 has been connected to the ECU 10. Alternatively, the rewriting operation may be initiated by operating the ECU.

[0041]

At step 41, the security release request part 21 of the rewriting device 11 transfers a signal indicative of a request for releasing security to the ECU 10. The ECU 10 responds to this signal to start an authentication process for confirming that the authorized rewriting device is connected thereto.

[0042]

An example of the authentication process is shown in FIG. 5. At step 51, the security release request part 21 of the rewriting device 11 requests the ECU 10 to transfer an arbitrary number R. In response to this, the authentication part 31 of the ECU 10 is invoked. The authentication part 31 invokes the random number generator 33 that generates random numbers. The authentication part 31 arbitrarily selects the number R from the random numbers generated by the random number generator 33, and transfers the number R to the rewriting device 11 (step 52). Alternatively, a different mechanism may be used to set

the arbitrary number R. The rewriting device 11 uses the security function f_1 already stored therein to determine the function value K1 of the function f_1 for the number R based on $K1 = f_1(R)$ (step 53).

[0043]

On the other hand, the authentication part 31 of the ECU 10 uses the security function f_2 stored in the rewritable ROM 16 to determine a function value K2 based on $K2 = f_2(R)$ (step 54). The security release request part 21 of the rewriting device 11 transfers the function value K1 to the ECU 10 (step 55). The authentication part 31 compares the function value K1 from the rewriting device 11 with the internally determined function value K2 (step 56), and if they are equal, determines that the rewriting device 11 is authentic. Subsequently, the authentication part 31 checks whether the engine start permission flag stored in the RAM 37 is a value of one (step 57). If the permission flag is one, this means that the engine start permission signal has been output from the anti-theft system 81, and a signal indicative of a permission of rewriting is transferred to the rewriting device 11 (step 58). Thus, the security feature needs to be released for rewriting data stored in the rewritable ROM, so that the current security functions f_1 and f_2 are used to release the security feature. With the anti-theft system mounted in the vehicle, the security feature for the memory rewriting system is released only if the anti-theft system has been released, thereby preventing an illegal driver from rewriting information relating to the rewriting system and the anti-theft system.

[0044]

Referring to FIG. 4 again, if the ECU authenticates the rewriting device 11, the process proceeds to step 42. The rewriting request part 23 of the rewriting device 11 transfers a signal indicative of a start of rewriting to the ECU 10, and the rewriting part 35 of the ECU 10 returns a start permission signal when ready for rewriting. At step 43, the rewriting device 11 transfers a request for shifting to a rewriting operation mode to the ECU 10, and then the rewriting part 35 of the ECU 10 executes a process for shifting to the rewriting operation mode. At step 44, the rewriting request part 23 queries the ECU 10 if the shift of the operation mode has completed. The rewriting part 35 transfers a signal indicative of a completion of the shift to the rewriting device 11 if the shift has been completed.

[0045]

At step 45, the rewriting request part 23 requests the security function f_2 stored in the rewritable ROM 16 to be deleted, and in response to this, the rewriting part 35 deletes the security function f_2 from the ROM 16.

[0046]

At this point, in the rewriting device 11, the new security function f_3 has been prepared as a new security function. The function f_3 has been provided by the data block assembling part 25 as serial data blocks for transmission to the ECU 10. The security function f_3 is typically created before the rewriting device 11 transfers the request for releasing security or the notification for starting of rewriting to the

ECU 10. This preparation for the new security function f_3, however, may be carried out immediately before the step 45.

[0047]

The new security function f_3 may be prepared, for example, by selecting one from a number of functions previously saved in the rewriting device 11. Alternatively, a user may create the new security function f_3 by manipulating the rewriting device 11.

[0048]

At step 46, the rewriting request part 23 transfers the first one of the data blocks representative of the new security function f_3 to the ECU 10 together with a signal indicative of a request for writing. The rewriting part 35 receives the data block from the rewriting device 11 and writes a partial program code included in the data block to the rewritable ROM 16. Once writing of the partial program code has been completed, the rewriting part 35 transfers a notification of the completion of writing to the rewriting device 11. In response to this, the rewriting device 11 transfers a next data block to the ECU 10. This step 46 is repeated until all the program code of the security function f_3 is written into the ROM 16.

[0049]

Once writing of all the program code has completed, the rewriting request part 23 transfers a request for releasing the rewriting operation mode to the ECU 10 (step 47). In response to this, the rewriting part 35 releases the rewriting operation mode. Since the rewriting device 11 has changed the security function stored in the ROM 16 to f_3, the function

used by the rewriting device 11 is also set to f_3 so that the security feature can subsequently be implemented by means of the security function f_3. After the new security function f_3 has been written to the ROM 16, the preceding security function f_1 may be deleted.

[0050]

FIG. 6 is a flow chart showing a process for releasing security executed by the rewriting device 11. At step 61, the rewriting device 11 requests a number R from the ECU 10. The rewriting device 11 subsequently receives the number R from the ECU 10 (step 62). Upon receiving the number R, the rewriting device 11 calculates the function value K1 for the number R using the security function f_1 already retained therein (step 63). Subsequently, the rewriting device 11 transfers the function value K1 to the ECU 10 (step 64).

[0051]

FIG. 7 is a flow chart showing a process for releasing security executed by the ECU 10. The ECU 10 receives the request for the number R from the rewriting device 11. Upon receiving the request, the ECU 10 sets the number R from random numbers (step 72) and transfers it to the rewriting device 11 (step 73). The ECU then calculates the function value K2 for the number R using the security function f_2 already retained therein (step 74).

[0052]

The ECU 10 receives the function value K1 from the rewriting device 11 (step 75) and compares the value K1 with the value K2 (step 76). If they are equal, the ECU 10 checks

whether the engine start permission flag is one (step 77). If the flag is one, the process proceeds to step 78 to set a rewriting permission flag, thereby indicating that the rewriting device 11 is permitted for rewriting. If the values are unequal at step 76 or the engine start permission flag is not set to a value of one at step 77, then the rewriting permission flag is set to zero (step 79) to indicate that the rewriting device is not permitted for rewriting and the process terminates.

[0053]

FIG. 8 is a flow chart of a process for rewriting executed by the rewriting device 11. At step 81, the rewriting device 11 transfers a request for rewriting to the ECU. The request may actually include the notification for a start of rewriting, the request for shifting to the rewriting operation mode, and the like, as shown in FIG. 4. Upon receiving a permission of rewriting provided by the ECU 10 in response to the request for rewriting (step 82), the rewriting device 11 creates data blocks of the new security function f_3 (step 83). The new security function f_3 can be arbitrarily created using the rewriting device 11 as described above. The rewriting device 11 then transfers the data blocks representative of the new security function f_3 to the ECU 10 (step 84).

[0054]

FIG. 9 is a flow chart showing a process for rewriting executed by the ECU. Upon receiving the request for rewriting from the rewriting device 11 (step 91), the ECU 10 checks whether the rewriting permission flag is set to one (step 92). If the

flag is set to one, which means that the rewriting device 11 has been proved to be authentic, then the ECU waits for the new security function f_3 transferred from the rewriting device 11. In fact, processes such as shifting to the rewriting operation mode or deletion of the current security function f_2 from the rewritable ROM as shown in FIG. 4 can be executed between steps 92 and 93.

[0055]

Subsequently, upon receiving the new security function f_3 (step 93), the ECU writes this function f_3 to the rewritable ROM. Thus, the security function f_2, which has been stored in the rewritable ROM, is rewritten with the new security function f_3.

[0056]

[Advantageous Effect of the Invention]

According to the invention set forth in claim 1, even if the security data for determining the presence of the permission of rewriting to prevent data stored in the memory of the vehicle controller from being rewritten illegally is divulged to a third party, the rewriting device can change the security data, thus preventing illegal rewriting to the memory from spreading.

[0057]

According to the invention set forth in claim 2, the program for rewriting the security data is stored in the unchangeable memory and is prevented from being tampered by a third party, thereby allowing the security data to be rewritten safely.

[0058]

According to the invention set forth in claim 3, the rewriting device can arbitrarily set a new security data, so that the new security data can be flexibly set without being divulged to any third person.

[0059]

According to the invention set forth in claim 4, the memory is rewritten only if the anti-theft system permits an operation for the vehicle, so that rewriting by an illegal driver is avoided, thus preventing information on the anti-theft system from being rewritten.

[Brief Description of the Drawings]

[Figure 1] A view showing an outline of a memory rewriting system according to one embodiment of the present invention.

[Figure 2] A block diagram showing an entire memory rewriting system according to one embodiment of the present invention.

[Figure 3] A view showing examples of a form of a ROM and a CPU of an ECU in a memory rewriting system according to one embodiment of the present invention.

[Figure 4] A view showing an operational procedure of a memory rewriting system according to one embodiment of the present invention.

[Figure 5] A view showing an authentication procedure executed by a memory rewriting system according to one embodiment of the present invention.

[Figure 6] A flow chart showing a process for releasing security executed by a rewriting device of a memory rewriting system according to one embodiment of the present invention.

[Figure 7] A flow chart showing a process for releasing security executed by an ECU of a memory rewriting system according to one embodiment of the present invention.

[Figure 8] A flow chart showing a process for rewriting executed by a rewriting device of a memory rewriting system according to one embodiment of the present invention.

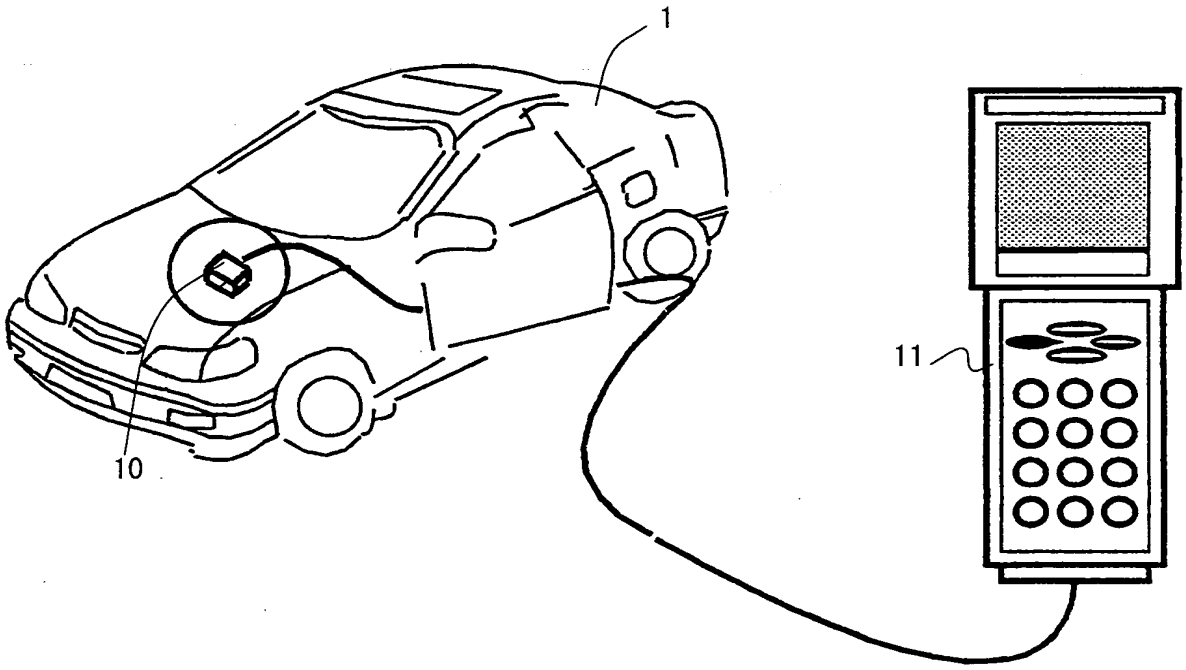
[Figure 9] A flow chart showing a process for rewriting executed by an ECU of a memory rewriting system according to one embodiment of the present invention.

[Explanations of Letters or Numerals]

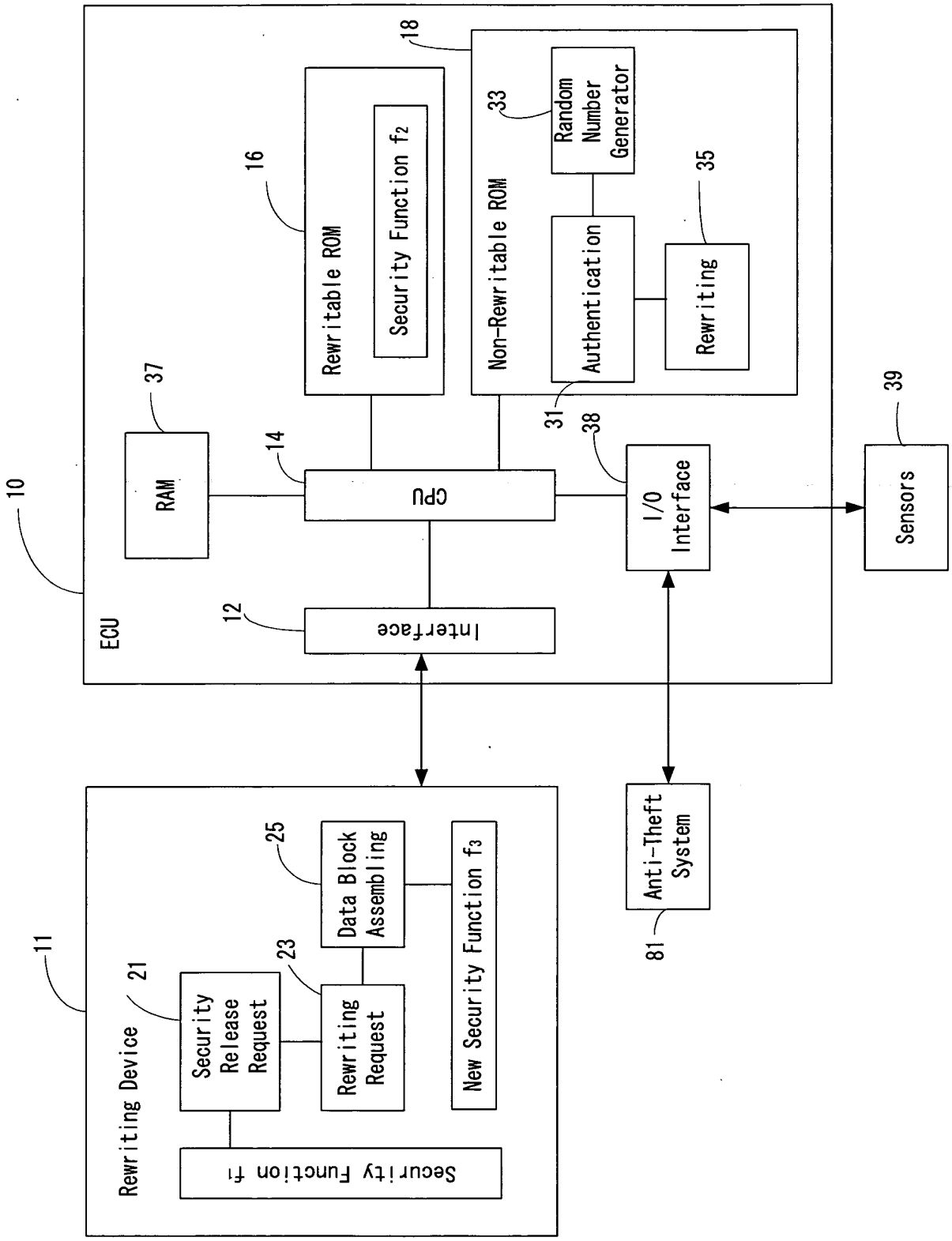
10	ECU	11	rewriting device
12	interface	14	CPU
16	rewritable ROM	18	non-rewritable ROM
81	anti-theft system		

[Document name] Drawings

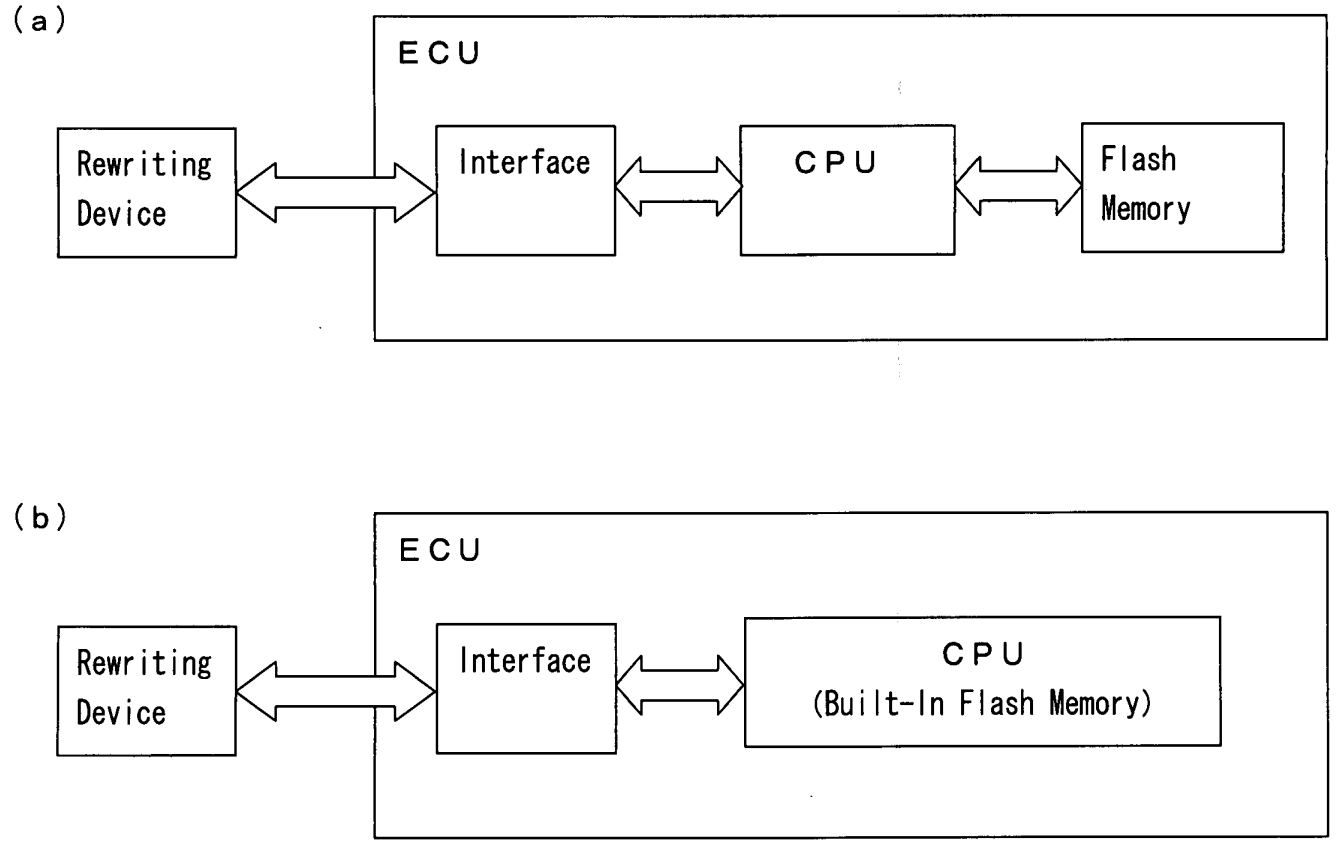
[Figure 1]



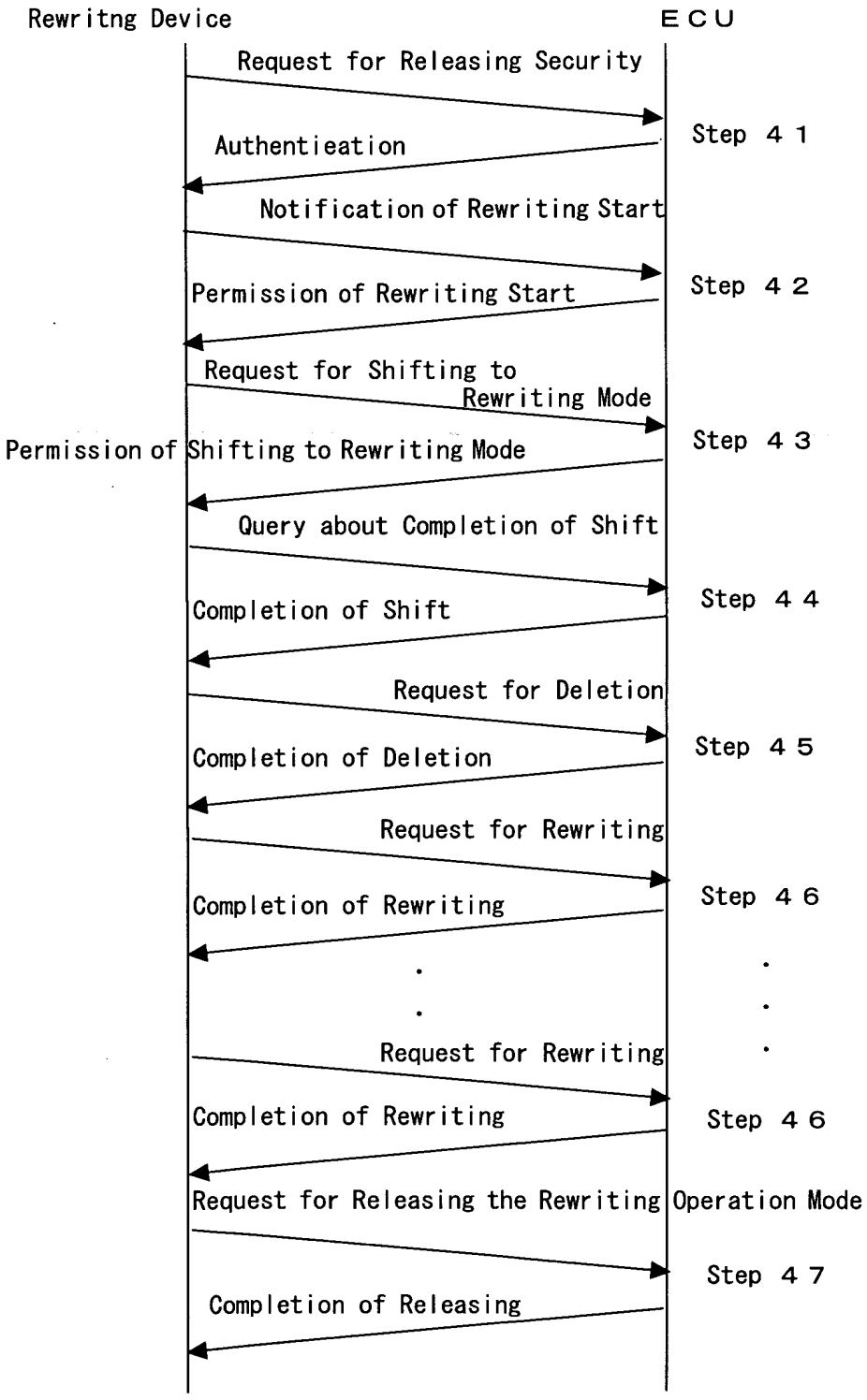
[Figure 2]



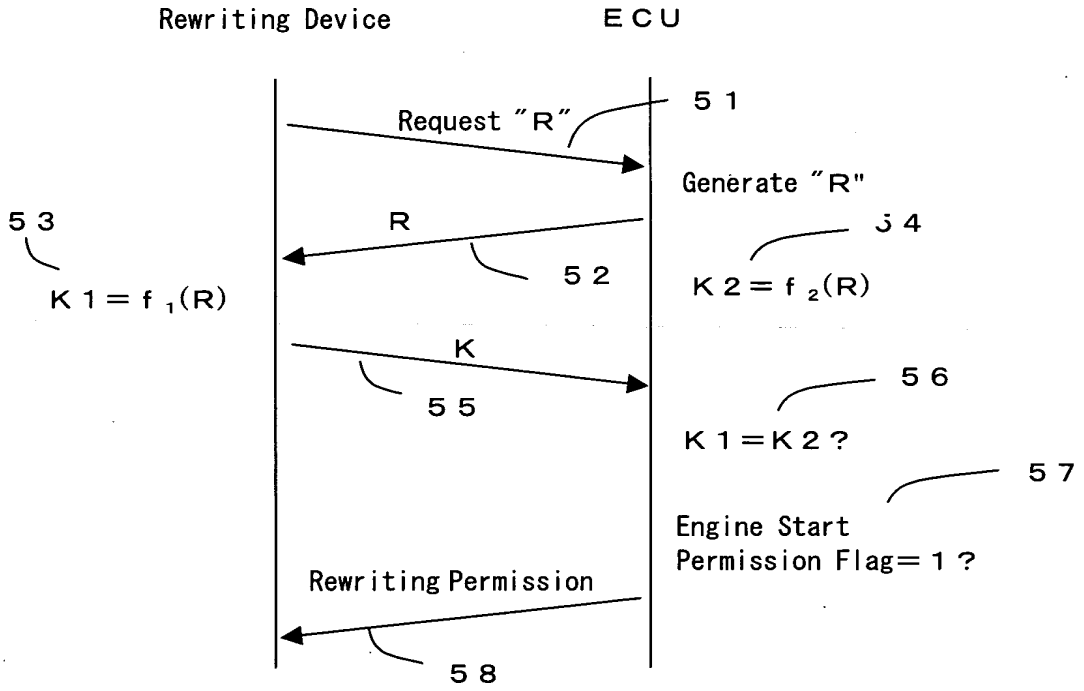
[Figure 3]



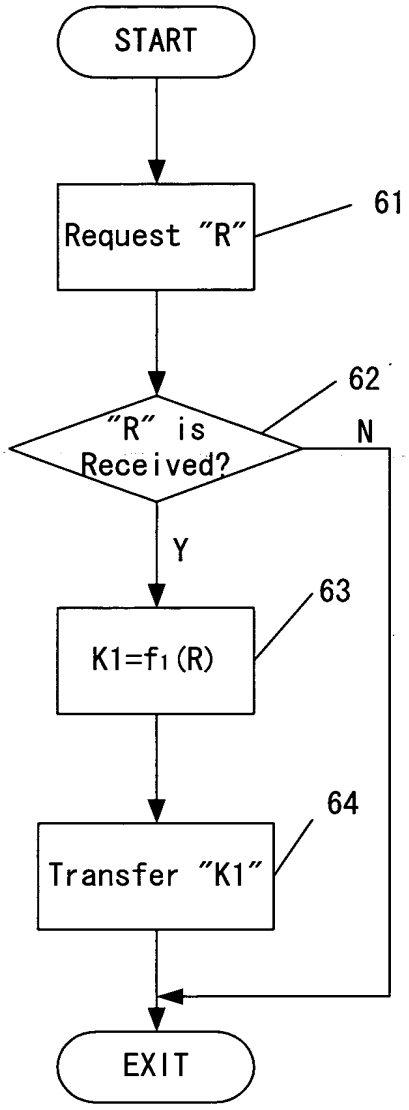
[Figure 4]



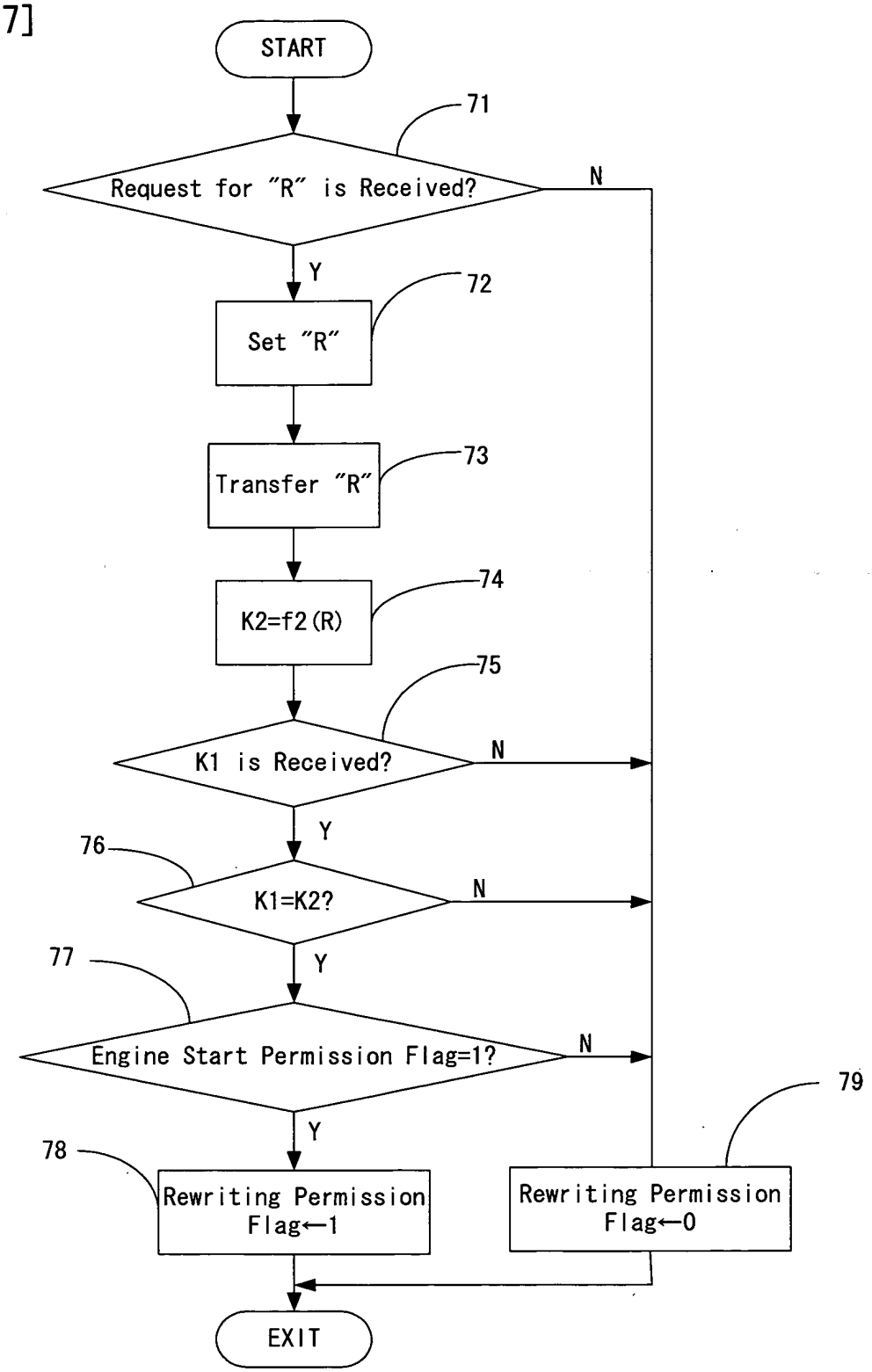
[Figure 5]



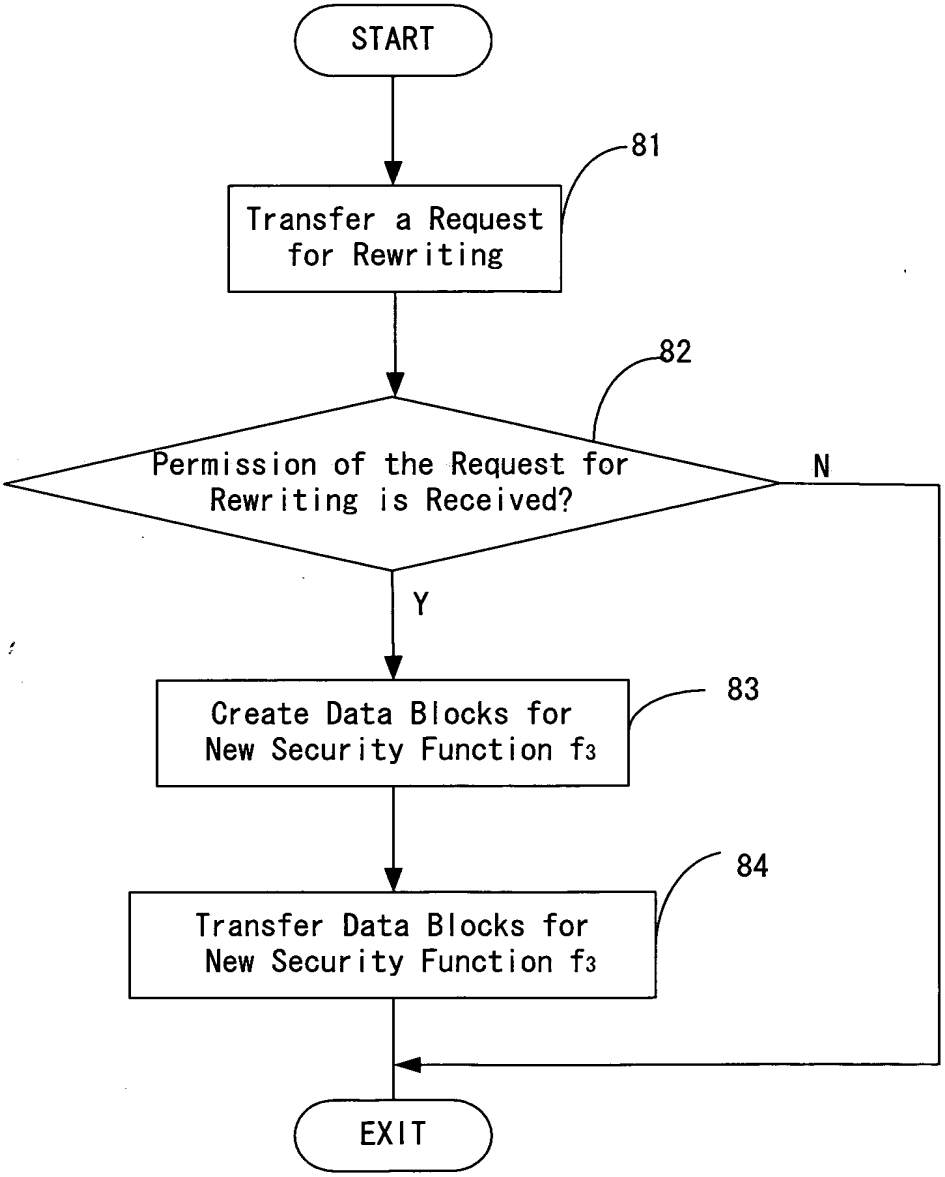
[Figure 6]



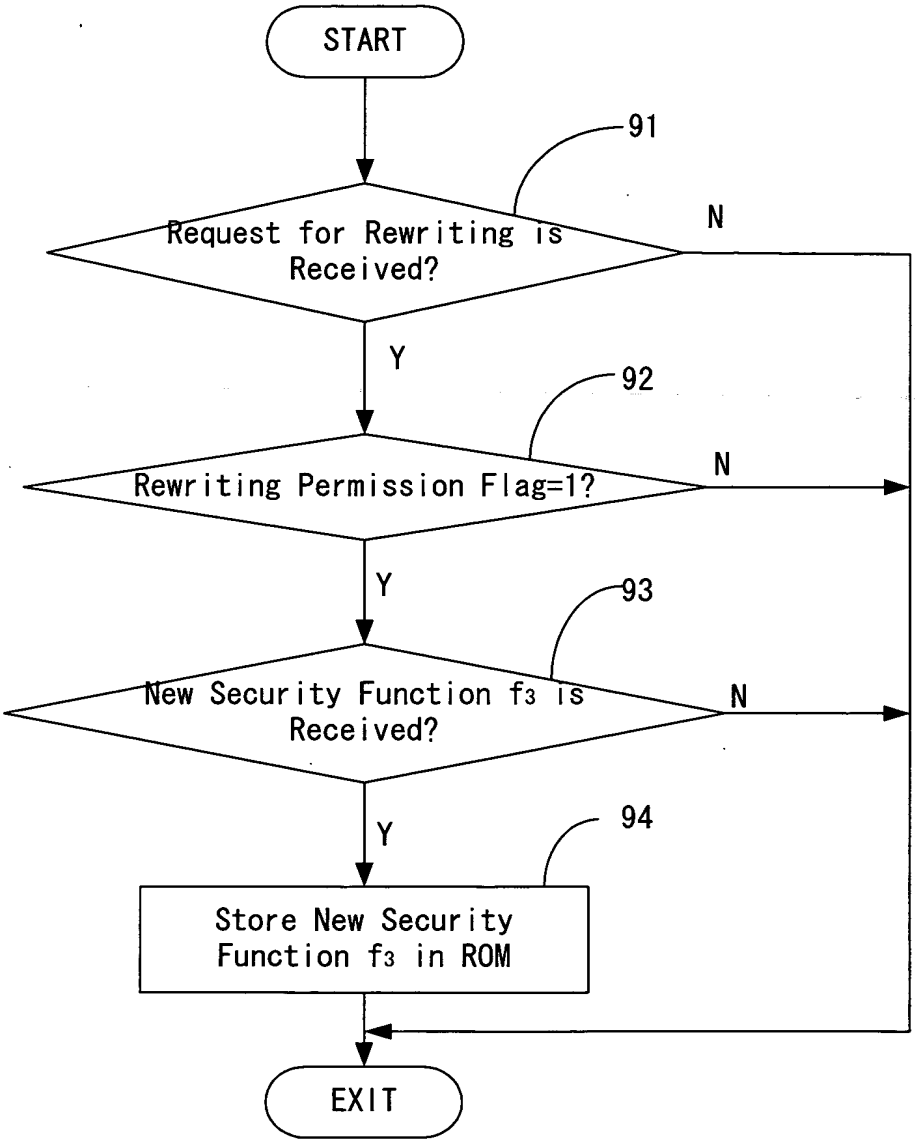
[Figure 7]



[Figure 8]



[Figure 9]



[Document name] Abstract

[Abstract]

[Problems to be Solved] This invention prevents illegal rewriting by enabling security data stored in a memory of a vehicle controller to be rewritten.

[Means to Solve the Problems] A rewriting system for a vehicle controller mounted in the vehicle controller, comprising a memory area from and to which data can be deleted and written and which stores first security data for determining the presence of a permission of rewriting to the memory area, a rewriting device for transferring second security data from an exterior to the vehicle controller, and a rewriting means mounted in the vehicle controller, for deleting the first security data and writing the second security data transferred from the rewriting device. Since the information for implementing a security feature for rewriting can be rewritten, the security feature can be recovered even if the information is divulged to a third party. Furthermore, the rewriting system can operate in cooperation with an anti-theft system.

[Selected Figure] Figure 2